

Reduced Functional Dependence Graphs and Their Applications

Xiaoli Xu[†], Satyajit Thakor[‡] and Yong Liang Guan[†]

[†]School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore

[‡]Institute of Network Coding, The Chinese University of Hong Kong, Shatin, Hong Kong

{xu0002li, eylguan}@e.ntu.edu.sg, thakor@inc.cuhk.edu.hk

Abstract—Functional dependence graphs (FDGs) are an important class of directed graphs that capture the functional dependence relationship among a set of random variables. FDGs are frequently used in characterizing and calculating network coding capacity bounds. However, the order of an FDG is usually much larger than the original network and the complexity of computing bounds grows exponentially with the order of an FDG. In this paper, we introduce graph pre-processing techniques which deliver reduced FDGs. These reduced FDGs are obtained from the original FDG by removing nodes that are not “essential”. We show that the reduced FDGs give the same capacity region/bounds obtained using original FDGs, but require much less computation. The application of reduced FDGs for algebraic formulation of scalar linear network coding is also discussed.

Index Terms—network coding capacity, linear programming bounds, functional dependence graphs, complexity reduction, algebraic formulation.

I. INTRODUCTION

Characterizing the capacity region of network coding is an important fundamental problem. For single session networks, it is well known that the capacity region is given by the max-flow bound [1]. However, the max-flow bound is no longer tight when multi-session networks are considered. It is shown in [6] that the exact capacity region for general networks can be written as a function of entropy region, Γ_n^* , and some constraints induced by the network topology. Unfortunately, characterization of Γ_n^* is still open for $n \geq 4$ and infinite number of information inequalities need to be considered [7]. An explicit outer bound in terms of the set of polymatroidal functions, Γ_n is referred as Linear Programming (LP) bound [8].

Functional dependence graphs (FDG) are a class of directed graphs that capture the functional dependence relationship among a set of random variables. FDGs are first used in [2] to establish conditional independence among random variables involved in a communication system which is useful to characterize bounds on the capacity of the communication system. Variants of FDGs are used in [3] and [5] to characterize computable outer bounds on multi-session network coding capacity. FDG plays an important role in studying the capacity for networks with edge capacity constraints. Besides the progressive d -separating edge-set bound [3] and the functional dependence bound [5] which are obtained using FDGs, many other capacity bounds e.g. [4], and the LP bound are closely

related with FDGs of the network. The random variables and the functional dependence constraints which are important in computing the LP bound of a given network are directly reflected by FDGs.

In this paper, we introduce graph pre-processing techniques to reduce the size of an FDG by removing edges and nodes that are not essential. The resulting FDG is referred as Reduced Functional Dependence Graph (FDG) which is of smaller size but preserves important properties of the original network. We show that the capacity of the original network can totally be determined by the reduced FDGs. As a result, reduced FDGs can be used to compute bounds on the network coding capacity of a given network with lesser computational resources. Moreover, removing nodes in the original FDGs is equivalent to removing the random variables involved in computing the LP bound of a given network. Hence the complexity of computing LP bound reduces exponentially using the notion of reduced FDGs.

Due to the similarities between an FDG and the directed line graph [9] of a given network, an FDG can also be used to construct the algebraic formulation of scalar linear network coding. With the proposed reduced FDG, the number of variables required for such formulation will be reduced and the complexity for computing the transition matrix can also be significantly reduced.

The rest of the paper is organized as follows. In section II, network model and formal definition of an FDG are given. Graph pre-processing techniques leading to reduced FDGs are given in section III. Applications of the main results are discussed in section IV, including complexity reduction for computing LP bound and complexity reduction for algebraic formulation of scalar linear network coding. Finally, the paper is concluded in section V.

II. BACKGROUND

A. Network Model

A network is represented by a directed acyclic graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where \mathcal{V} is the set of nodes and \mathcal{E} is the set of edges. For an edge $E = (V, U) \in \mathcal{E}$, define $\text{Tail}(E) = V$ and $\text{Head}(E) = U$. For $V \in \mathcal{V}$, the set of edges entering into V and leaving V are denoted by $\text{In}(V)$ and $\text{Out}(V)$ respectively. To simplify the description, we also define the set of edges entering into and leaving an edge E as $\text{In}(E) = \{E' : \text{Head}(E') = \text{Tail}(E)\}$ and $\text{Out}(E) = \{E' : \text{Tail}(E') =$

$\text{Head}(E)\}$ respectively. It is easy to see that if $E = (V, U)$, $\text{In}(E) = \text{In}(V)$ and $\text{Out}(E) = \text{Out}(U)$.

Let $\mathcal{S} = \{1, \dots, |\mathcal{S}|\}$ denote the set of independent information sources available at some nodes (called source nodes) in a network via mapping $a : \mathcal{S} \mapsto \mathcal{V}$. The sources are demanded by some nodes in a network called sink nodes. A set of sources demanded by a given sink node is described by the mapping $b : \mathcal{V} \mapsto \mathcal{P}(\mathcal{S})$ where, $\mathcal{P}(\mathcal{A})$ is the power set of the set \mathcal{A} . Let \mathcal{T} denote the set of sink nodes in the network and thus the set of sources demanded by the sink $T \in \mathcal{T}$ is $b(T)$. If each source is demanded by exactly one sink, the network is called multiple-unicast network. Without loss of generality, we assume that $\text{In}(\mathcal{S}) = \text{Out}(\mathcal{T}) = \emptyset, \forall S \in \mathcal{S}, T \in \mathcal{T}$. For $E \in \mathcal{E}$, C_E denotes the maximal rate that can be conveyed through the link E .

Definition 1: For a given network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, an information rate tuple $\mathbf{R} = (R_S : S \in \mathcal{S})$ is achievable if there exists a network code $\mathbf{C} = \{f_E : E \in \mathcal{E}, g_T : T \in \mathcal{T}\}$ of block length n , defined by

- For all $S \in \mathcal{S}, E \in \text{Out}(S)$, local encoding functions $f_E : \mathcal{Y}_S^{(n)} \rightarrow \mathcal{U}_E^{(n)}$,
- For all $V \in \mathcal{V} \setminus (\mathcal{S} \cup \mathcal{T}), E \in \text{Out}(V)$ local encoding functions $f_E : \prod_{E' \in \text{In}(V)} \mathcal{U}_{E'}^{(n)} \rightarrow \mathcal{U}_E^{(n)}$,
- and for all $T \in \mathcal{T}$, decoding functions $g_T : \prod_{E \in \text{In}(T)} \mathcal{U}_E^{(n)} \rightarrow \prod_{S \in b(T)} \mathcal{Y}_S^{(n)}$

such that

$$\lim_{n \rightarrow \infty} n^{-1} \log |\mathcal{Y}_S^{(n)}| = \lim_{n \rightarrow \infty} n^{-1} H(Y_S^{(n)}) \geq R_S$$

$$\lim_{n \rightarrow \infty} n^{-1} H(U_E^{(n)}) \leq \lim_{n \rightarrow \infty} n^{-1} \log |\mathcal{U}_E^{(n)}| \leq C_E$$

and for all $T \in \mathcal{T}$

$$\lim_{n \rightarrow \infty} \Pr(g_T(U_E^{(n)} : E \in \text{In}(T)) \neq (X_S^{(n)} : S \in b(T))) = 0.$$

The closure of the set of all achievable rate tuples for a given network is called the capacity of the network.

B. Functional Dependence Graph

Given a network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with sets of random variables $\{Y_S : S \in \mathcal{S}\}$ and $\{U_E : E \in \mathcal{E}\}$ representing the information generated by sources and carried by edges respectively, a valid network code for the network satisfies the following constraints.

$$R_S \leq H(Y_S) \quad (1)$$

$$H(Y_S) = \sum_{S \in \mathcal{S}} H(Y_S) \quad (2)$$

$$H(U_{\text{Out}(S)} | Y_S) = 0, S \in \mathcal{S} \quad (3)$$

$$H(U_{\text{Out}(V)} | U_{\text{In}(V)}) = 0, V \in \mathcal{V} \setminus (\mathcal{S} \cup \mathcal{T}) \quad (4)$$

$$H(Y_{b(T)} | U_{\text{In}(T)}) = 0, T \in \mathcal{T} \quad (5)$$

$$H(U_E) \leq C_E, E \in \mathcal{E} \quad (6)$$

Note that constraint (2) specifies the independence of the sources, (3) and (4) give encoding constraints, (5) corresponds to the decoding requirements, (6) means that the entropy of variables carried by any link cannot exceed the link capacity.

Using the encoding and decoding constraints (3)-(5) for a given network \mathcal{G} (i.e., local functional dependence constraints at nodes), an FDG can be constructed for the set of source and edge random variables. FDG is defined in [5] as:

Definition 2 (Functional dependence graph [5]): Let $\bar{\mathcal{V}}$ be a set of random variables. A directed graph $\bar{\mathcal{G}} = (\bar{\mathcal{V}}, \bar{\mathcal{E}})$ is called a functional dependence graph for $\bar{\mathcal{V}}$ if and only if for all $V \in \bar{\mathcal{V}}$

$$H(V | \{U : (U, V) \in \bar{\mathcal{E}}\}) = 0. \quad (7)$$

$\bar{\mathcal{G}}$ is a directed cyclic graph and of the order $|\bar{\mathcal{V}}| = |\mathcal{E}| + |\mathcal{S}|$. $\bar{\mathcal{E}}$ is determined by the dominance relationship defined in (3)-(5). For $V \in \bar{\mathcal{V}}$, let $\text{Down}(V) \triangleq \{V' : (V, V') \in \bar{\mathcal{E}}\}$ be the set of immediate downstream nodes (also known as children) of V and $\text{Up}(V) \triangleq \{V' : (V', V) \in \bar{\mathcal{E}}\}$ be the set of immediate upstream nodes (also known as parents) of V . Thus, for $S \in \mathcal{S}$, $\text{Down}(Y_S) = U_{\text{Out}(S)}$ and if $b(T) = S$, $\text{Up}(Y_S) = U_{\text{In}(T)}$. Moreover, for $E \in \mathcal{E}$, $\text{Down}(U_E) = U_{\text{Out}(E)}$ and $\text{Up}(U_E) = U_{\text{In}(E)}$. Provided that the communication network \mathcal{G} is directed acyclic, every node $Y_S, S \in \mathcal{S}$ in $\bar{\mathcal{G}}$ must be a member of a cycle due to the decoding constraints (5).

III. MAIN RESULTS

A. Reduced Functional Dependence Graph

For a given network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, if it is a connected graph, $|\mathcal{E}| \geq |\mathcal{V}| - 1$. Therefore the order of an FDG of the given network, $|\bar{\mathcal{V}}| = |\mathcal{E}| + |\mathcal{S}|$, is usually much larger than the original network. As the complexity of computing network capacity bounds usually grows exponentially with the size of an FDG, it is desirable to reduce the size of it.

Source random variables $Y_S, S \in \mathcal{S}$ may be regarded as primary random variables introducing randomness into the network. Edge random variables $U_E, E \in \mathcal{E}$ may be regarded as secondary random variables since all edge random variables are function of the source random variables. In this section we focus on eliminating certain edge random variables (and hence corresponding nodes in FDGs) such that the capacity of the network remains unchanged.

Definition 3: Given a network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, let $E = (V, U) \in \mathcal{E}$, the reduced network $\mathcal{G} \setminus E = (\mathcal{V}', \mathcal{E}')$ is defined by: $\mathcal{V}' = \mathcal{V}$ and $\mathcal{E}' = \mathcal{E} \setminus E$. Moreover, the information carried by edges in $\text{In}(E)$ are made available to both node V and U , i.e. for $K \in \text{In}(E)$, $\text{Head}(K) = \{U, V\}$.

The procedure for obtaining $\mathcal{G} \setminus E$ from \mathcal{G} is shown in Fig.1. Note that the reduced network $\mathcal{G} \setminus E$ is no longer a traditional network since some edges have more than one destination nodes. However, the definition of capacity still applies.

Although the transformation from \mathcal{G} to $\mathcal{G} \setminus E$ makes the network look more complex, it reflects a reduction in their corresponding FDG sub-networks (referred as $\bar{\mathcal{G}}$ and $\bar{\mathcal{G}} \setminus U_E$ respectively) as shown in Fig.2.

Definition 4: Given an FDG $\bar{\mathcal{G}} = (\bar{\mathcal{V}}, \bar{\mathcal{E}})$, a reduced FDG $\bar{\mathcal{G}} \setminus U_E = (\bar{\mathcal{V}}', \bar{\mathcal{E}}')$ is given by $\bar{\mathcal{V}}' \triangleq \bar{\mathcal{V}} \setminus U_E$ and

$$\bar{\mathcal{E}}' \triangleq \{\bar{\mathcal{E}} \setminus \{\bar{E} \in \bar{\mathcal{E}} : \text{Head}(\bar{E}) = U_E \text{ or } \text{Tail}(\bar{E}) = U_E\}\} \cup \{(A, B) : A \in \text{Up}(U_E), B \in \text{Down}(U_E)\}.$$

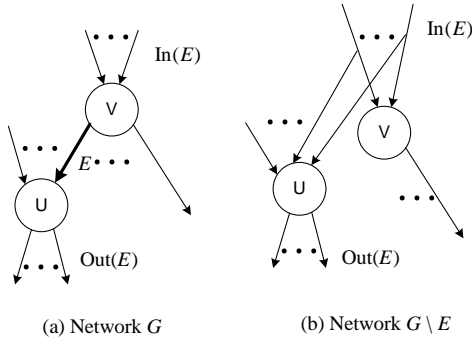


Fig. 1. Construct new network by removing edge E

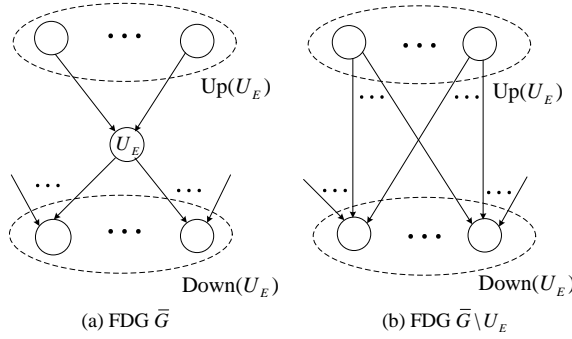


Fig. 2. Construct new FDG by removing U_E

Theorem 1: Network \mathcal{G} and $\mathcal{G} \setminus E$ have the same capacity if the following conditions are satisfied:

- 1) $\text{Tail}(E) \notin \mathcal{S}$
- 2) $C_E \geq \sum_{E' \in \text{In}(E)} C_{E'}$

Proof: Assume that an information rate tuple $\mathbf{R} = (R_S : S \in \mathcal{S})$ is achievable in the original network \mathcal{G} . According to Definition 1, there exists a network code \mathbf{C} , composed of a set of encoding functions $\{f_E : E \in \mathcal{E}\}$ and a set of decoding functions $\{g_T : T \in \mathcal{T}\}$, with which the communication requirement is satisfied. Given \mathbf{C} , we can easily construct a network code \mathbf{C}' for the new network $\mathcal{G} \setminus E$ via the following modification:

- excluding the encoding function for the removed edge E
- if $\text{Head}(E) \notin \mathcal{T}$, for edge $K \in \text{Out}(E)$, let $\mathcal{X} \triangleq \text{In}(K) \setminus E$ be the set of incoming edges to the edge K except E . Replacing f_K with a new composite function $f'_K = f_K(f_E(\prod_{E' \in \text{In}(E)} \mathcal{U}_{E'}^{(n)}), \prod_{X \in \mathcal{X}} \mathcal{U}_X^{(n)})$
- if $\text{Head}(E) = T \in \mathcal{T}$, let $\mathcal{X} = \text{In}(T) \setminus E$. Changing g_T to $g'_T = g_T(f_E(\prod_{E' \in \text{In}(E)} \mathcal{U}_{E'}^{(n)}), \prod_{X \in \mathcal{X}} \mathcal{U}_X^{(n)})$

With above modification, if $\text{Head}(E) \notin \mathcal{T}$, $\mathbf{C}' = (\mathbf{C} \setminus \{f_E, \{f_K, K \in \text{In}(E)\}\}) \cup \{f'_K\}$ and if $\text{Head}(E) = T \in \mathcal{T}$, $\mathbf{C}' = (\mathbf{C} \setminus \{f_E, g_T\}) \cup \{g'_T\}$. Note that the encoding function f'_K defines a mapping: $\prod_{X \in \mathcal{X}} \mathcal{U}_X^{(n)} \cdot \prod_{E' \in \text{In}(E)} \mathcal{U}_{E'}^{(n)} \rightarrow \mathcal{U}_K^{(n)}$ which produces the same output as f_K in the original network code \mathbf{C} . Similar arguments hold for the decoding function g'_T if $\text{Head}(E) = T \in \mathcal{T}$. Therefore, the rate tuple $\mathbf{R} = (R_S : S \in \mathcal{S})$ is achievable in $\mathcal{G} \setminus E$ with the network code \mathbf{C}' .

On the other hand, assume that a rate tuple $\mathbf{R} = (R_S : S \in \mathcal{S})$ is achievable in $\mathcal{G} \setminus E$. There exists a network code \mathbf{C}' , with which the communication requirement can be satisfied in $\mathcal{G} \setminus E$. Based on \mathbf{C}' , we can construct the network code \mathbf{C} for the original network if the conditions stated in this theorem are satisfied.

Condition 1) states $\text{Tail}(E) \notin \mathcal{S}$ and Condition 2) states that capacity of edge E is larger than or equal to sum capacity of all edges coming into node V . Combined both conditions, we know that edge E is capable of forwarding all received data without any compression. Therefore, we can set the encoding function at edge E as a simple replication, i.e. $\mathcal{U}_E^{(n)} = f_E(\prod_{E' \in \text{In}(E)} \mathcal{U}_{E'}^{(n)}) = \prod_{E' \in \text{In}(E)} \mathcal{U}_{E'}^{(n)}$. Thus, the network code $\mathbf{C} = \mathbf{C}' \cup f_E$. Note that this construction does not violate any encoding constraint for downstream edges of E since the same information is available to them for encoding as in the reduced FDG. Therefore, all the variables in the original network are the same as that in the reduced network and thus decoding requirements are satisfied. ■

Due to the correspondence of network and its FDG, Theorem 1 can be applied to reduce the size of FDG. Before introducing the corollary of Theorem 1 in FDG, the definition of redundant random variable is made.

Definition 5: A node U_E (corresponding to the edge random variable U_E) in a functional dependence graph $\bar{\mathcal{G}} = (\bar{\mathcal{V}}, \bar{\mathcal{E}})$ is called *redundant* if the capacity region of the reduced FDG $\bar{\mathcal{G}} \setminus U_E$ is the same as the original FDG $\bar{\mathcal{G}}$.

Corollary 1: A node $U_E \in (\bar{\mathcal{V}} \setminus Y_S)$ is redundant if the following conditions are satisfied:

- 1) $Y_S \cap \text{Up}(U_E) = \emptyset$
- 2) $C_E \geq \sum_{A: U_A \in \text{Up}(U_E)} C_A$

Corollary 1 follows from Theorem 1 and Definition 5. A simple extension of Corollary 1 from single edge variable to a group of edge variables gives the following corollary.

Corollary 2: A set of nodes $U_{\mathcal{X}} \subset (\bar{\mathcal{V}} \setminus Y_S)$ is redundant if the following conditions are satisfied:

- 1) $\text{Up}(U_{E_1}) = \text{Up}(U_{E_2}), \forall E_1, E_2 \in \mathcal{X}$
- 2) $\text{Down}(U_{E_1}) = \text{Down}(U_{E_2}), \forall E_1, E_2 \in \mathcal{X}$
- 3) $Y_S \cap \text{Up}(U_{\mathcal{X}}) = \emptyset$ and $\sum_{E \in \mathcal{X}} C_E \geq \sum_{U_A \in \text{Up}(U_{\mathcal{X}})} C_A$

The proof of Corollary 2 follows by treating the set of variables $U_{\mathcal{X}}$ as a single “super-variable”. Note that the redundant variables identified by Corollary 1 and Corollary 2 correspond to the edges in original network where network coding is not necessary since they have sufficient capacity to forward all received information to their children.

In practise, most networks use packet-based transmissions. Therefore, we can assume unit-edge capacity. The edges that have large capacity (convey multiple packets) can be represented by multiple parallel edges of unit capacity. Based on this assumption, Corollary 1 and Corollary 2 can be simplified as follows.

Corollary 3: For $U_E \in (\bar{\mathcal{V}} \setminus Y_S)$, it is redundant if $|\text{Up}(U_E)| = 1$ and $\text{Up}(U_E) \cap Y_S = \emptyset$

Corollary 4: Let $U_{\mathcal{X}} \subset (\bar{\mathcal{V}} \setminus Y_S)$, they are redundant if the following conditions are satisfied.

- 1) $\text{Up}(U_{E_1}) = \text{Up}(U_{E_2}), \forall E_1, E_2 \in \mathcal{X}$
- 2) $\text{Down}(U_{E_1}) = \text{Down}(U_{E_2}), \forall E_1, E_2 \in \mathcal{X}$
- 3) $\text{Up}(U_{\mathcal{X}}) \cap Y_S = \emptyset$ and $|\text{Up}(U_{\mathcal{X}})| \leq |U_{\mathcal{X}}|$

Example 1: An example of reduced FDG is shown in Fig.3. The network in Fig.3(a) is constructed in [10] and each edge in the network has unit capacity. Fig.3(c) gives the FDG constructed based on Definition 2. This FDG can be reduced to the one shown in Fig.3(b) based on Corollary 3.

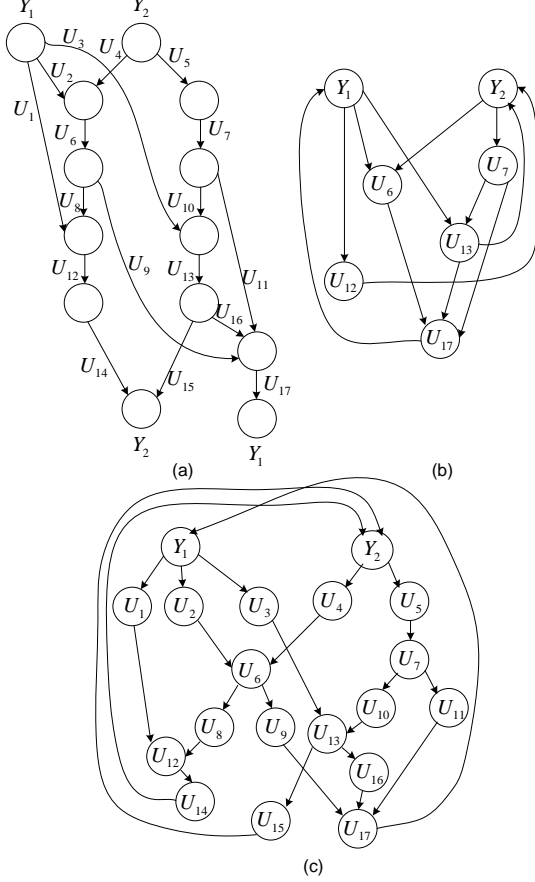


Fig. 3. (a)A network in [10] (b)The Reduced FDG (proposed) (c)The conventional FDG

B. Further Reduction of FDG

It has been proved in [11] that linear network coding is not sufficient to achieve the capacity for general networks. However, in most practical scenarios, linear network coding is preferred due to its simplicity. With linear network coding, the data carried by an edge E is a linear combination of input data available at $\text{Head}(E)$. Without loss of generality, we may assume that each edge has unit capacity and the encoding function at edge E is described by [9]:

$$f_E : U_E = \sum_{E' : E' \in \text{In}(E)} \varepsilon_{E,E'} U_{E'}$$

where the coefficients $\varepsilon_{E,E'}$ are elements chosen from designed alphabet and $H(U_E) \leq 1, \forall E \in \mathcal{E}$.

Theorem 2: Assume each edge in the network has unit capacity, then the network \mathcal{G} and $\mathcal{G} \setminus E$ (refer to Fig.1) have the same linear network coding capacity if one of the following conditions is satisfied:

- 1) $\text{Tail}(E) \notin \mathcal{S}$ and $|\text{In}(E)| = 1$
- 2) $\text{Head}(E) \notin \mathcal{T}$ and $|\text{Out}(E)| = 1$

Proof: As shown in the proof of Theorem 1, if a rate tuple $\mathbf{R} = (R_S : S \in \mathcal{S})$ is achievable in the \mathcal{G} , it is always achievable in $\mathcal{G} \setminus E$ with the same linear network code after simple modification.

On the other hand, if a rate tuple \mathbf{R} is achieved in $\mathcal{G} \setminus E$ with a linear network code \mathbf{C}' , we will show that a linear network code \mathbf{C} can be constructed for the original network \mathcal{G} with given conditions.

For $K \in \text{Out}(E)$, let $\mathcal{X} = \text{In}(K) \setminus E$. Since \mathbf{C}' is a linear network code, the encoding function for edge K is of the form: $f_K : U_K = \sum_{E' \in \text{In}(K)} \varepsilon_{E',K} U_{E'} = \sum_{E' \in \text{In}(E)} \varepsilon_{E',K} U_{E'} + \sum_{X \in \mathcal{X}} \varepsilon_{X,K} U_X$. If condition 1) is satisfied, \mathbf{C} is constructed from \mathbf{C}' by letting edge E perform simple forwarding, i.e. $f_E : U_E = U_{\text{In}(E)}$. If condition 2) is satisfied, the encoding function for edge E can be set to $f_E : U_E = \sum_{E' \in \text{In}(E)} \varepsilon_{E',K} U_{E'}$. For the edge $K = \text{Out}(E)$, the encoding function is changed to $f_K : U_K = U_E + \sum_{X \in \mathcal{X}} \varepsilon_{X,K} U_X$. For either case, the constructed code \mathbf{C} achieves the same rate tuple $\mathbf{R} = (R_S : S \in \mathcal{S})$ since all edges carry the same information as that in $\mathcal{G} \setminus E$. ■

Applying Theorem 2 in FDG results further reduction described by the following corollary.

Corollary 5: When linear network coding capacity is interested, a node $U_E \in (\tilde{\mathcal{V}} \setminus Y_S)$ is redundant if one of the following conditions is satisfied:

- 1) $|\text{Up}(U_E)| = 1$ and $Y_S \cap \text{Up}(U_E) = \emptyset$
- 2) $|\text{Down}(U_E)| = 1$ and $Y_S \cap \text{Down}(U_E) = \emptyset$

Note that the extension to group nodes is straightforward and thus omitted here due to space limitation.

Example 2: Consider the network shown in Fig.4(a) which appeared in [12] and has unit edge capacity. The corresponding FDG of this network is shown in Fig.4(b). By Corollary 1, it is reduced to 4(c) and it can be further reduced to the one in Fig.4(d) according to Corollary 5 if only linear network coding capacity is interested. With the reduced FDG, it is easy to show that $H(Y_1, Y_2 | U_8) = 0$. Thus the sum capacity that can be achieved with linear network coding is bounded by: $R_1 + R_2 \leq H(U_8) \leq 1$, which is the same as the capacity achieved by routing.

IV. APPLICATIONS OF REDUCED FDG

Although the concept of reduced FDG is very simple and the procedures to obtain reduced FDG are straightforward, the range of its application is wide and two of them are discussed in this section.

A. Complexity Reduction for Computing LP Bound

The Linear Programming (LP) bound is the set of all rate tuples satisfying (1)-(6) together with the basic inequalities.

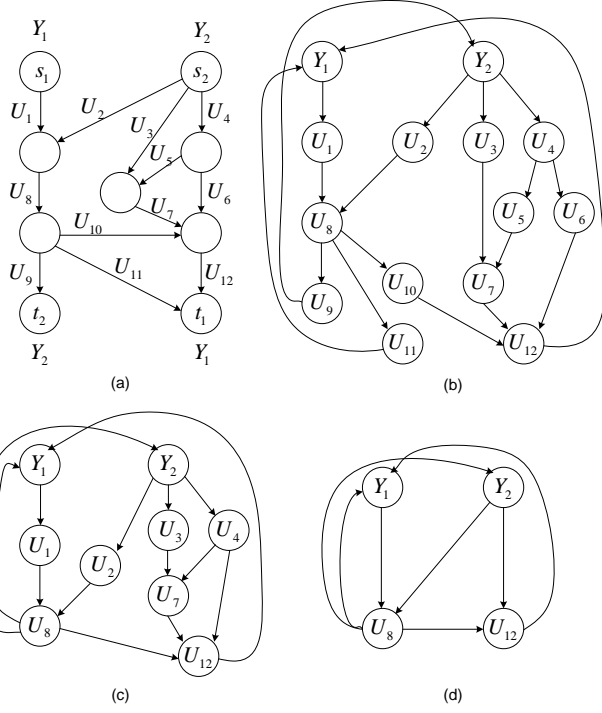


Fig. 4. (a)A network taken from [12] (b)The conventional FDG (c)The reduced FDG (proposed) (d)The further reduced FDG

The basic inequalities are non-negativity of entropy, conditional entropy and conditional mutual information. The set of the elemental basic inequalities

$$H(A|\{Y_S, U_E\} \setminus A) \geq 0, A \in \{Y_S, U_E\} \quad (8)$$

$$I(A; B|C) \geq 0, A \neq B \neq \emptyset, A, B \in \{Y_S, U_E\}, \\ C \subseteq \{Y_S, U_E\} \setminus \{A, B\} \quad (9)$$

represents (implies) all Shannon-type inequalities for the random variables Y_S, U_E and is minimal [8]. Note that the constraints (1)-(9) are linear and hence the LP bound can be computed by solving the linear program

$$\max \sum_{S \in \mathcal{S}} w_S H(Y_S) \text{ subject to } (2) - (6), (8) - (9) \quad (10)$$

where w_S is any non-negative constant for source S called weight coefficient.

For every set of chosen weight, the dimension of this optimization problem is $2^N - 1$, where $N = |\mathcal{E}| + |\mathcal{S}|$ and the number of constraints is $N + \binom{N}{2} 2^{N-2} + 1 + 2|\mathcal{E}| + |\mathcal{T}|$, including $N + \binom{N}{2} 2^{N-2}$ elemental information inequalities, 1 equality representing source independence, $|\mathcal{E}|$ equalities for encoding requirements, another $|\mathcal{E}|$ inequalities capturing the edge capacity constraints and $|\mathcal{T}|$ equalities stating the decoding requirements.

Note that the dimension of this optimization problem grows exponentially with the order of its FDG, N . Therefore, any reduction of FDG size results in exponential reduction of the problem dimension. The reduction in the number of constraints

is more significant since it grows even faster with the size of FDG.

Example 3: (Butterfly Network): Consider the Butterfly network shown in Fig.5(a). If the original FDG shown in Fig.5(b) is used. The number of variables involved is $N = 9$ and the dimension of the linear program is 511. A matrix representing the constraint set is of size 4634×511 . With the reduced FDG shown in Fig.5(c), N reduces to 5, thus the problem dimension and the constraint size reduces to 31 and 91×31 respectively. The overall complexity reduction exceeds 99%. Combining the reduced FDG with other complexity reduction techniques shown in [13], the dimension of the LP bound for Butterfly network can be further reduced to 8.

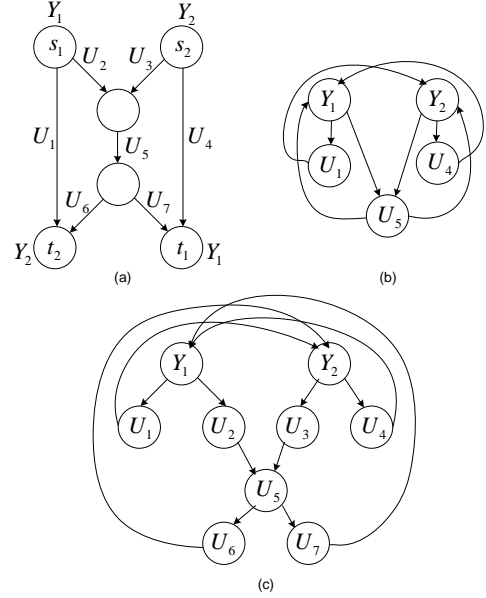


Fig. 5. (a)Butterfly Network (b)The reduced FDG (c)The conventional FDG

B. Complexity Reduction in Algebraic Formulation of the Scalar Network Coding

The directed line graph is introduced in [9] to construct an algebraic formulation of the scalar network coding. For a network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, let its line graph be denoted by $\tilde{\mathcal{G}} = (\tilde{\mathcal{V}}, \tilde{\mathcal{E}})$ and its FDG be denoted by $\bar{\mathcal{G}} = (\bar{\mathcal{V}}, \bar{\mathcal{E}})$. Each $V \in \tilde{\mathcal{V}}$ corresponds to an edge $E \in \mathcal{E}$ and $(V_1, V_2) \in \tilde{\mathcal{E}}$ if and only if $\text{Head}(V_1) = \text{Tail}(V_2)$ in \mathcal{G} . Compare it with the definition of FDG, it is not difficult to conclude that $\tilde{\mathcal{G}}$ is a subgraph of $\bar{\mathcal{G}}$. Therefore, $\tilde{\mathcal{G}}$ can be used to construct the algebraic formulation as well.

Following the similar notation used in [9], let A be the matrix that maps the source to its out-edges, F be adjacent matrix of $\tilde{\mathcal{G}}$ and B be the matrix that maps in-edges of the receiver to decoded symbols. The transition matrix of the network is formulated as $M = A(I - F)^{-1}B$ [9]. The total number of variables involved in this formulation is $|\tilde{\mathcal{E}}|$ and adjacent matrix F is of the size $|\tilde{\mathcal{V}}| \times |\tilde{\mathcal{V}}|$. Let Δ_V and Δ_E denote the amount of nodes and edges removed from the

$$M = A(I - F)^{-1}B = \begin{bmatrix} \alpha_2\beta_2\gamma_5 & \alpha_2\beta_1\gamma_3 + \alpha_2\beta_2\gamma_4 & \alpha_1\gamma_1 + \alpha_2\beta_1\gamma_2 \\ \alpha_4\gamma_6 + \alpha_3\beta_2\gamma_5 & (\alpha_3\beta_1 + \alpha_4\beta_3)\gamma_3 + \alpha_3\beta_2\gamma_4 & (\alpha_3\beta_1 + \alpha_4\beta_3)\gamma_2 \\ \alpha_5\gamma_6 + \alpha_6\gamma_5 & \alpha_5\beta_3\gamma_3 + \alpha_6\gamma_4 & \alpha_5\beta_3\gamma_2 \end{bmatrix} \quad (11)$$

original FDG respectively. Therefore, with the reduced FDG, the variables involved in the formulation is reduced by Δ_E and the reduction in complexity for computing the transition matrix is proportional to Δ_V .

Example 4: Fig.6(a) shows a network taken from [11] that is solvable only over fields with characteristic 2 and Fig.6(b) shows the corresponding reduced FDG. By labeling each edge in the reduced FDG, we can formulate the scalar linear network coding as:

$$A = \begin{bmatrix} \alpha_1 & \alpha_2 & 0 & 0 & 0 \\ 0 & \alpha_3 & \alpha_4 & 0 & 0 \\ 0 & 0 & \alpha_5 & 0 & \alpha_6 \end{bmatrix}$$

$$B^T = \begin{bmatrix} 0 & 0 & \gamma_6 & 0 & \gamma_5 \\ 0 & 0 & 0 & \gamma_3 & \gamma_4 \\ \gamma_1 & 0 & 0 & \gamma_2 & 0 \end{bmatrix}, F = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \beta_1 & \beta_2 \\ 0 & 0 & 0 & \beta_3 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

The transition matrix is given in (11). By setting the diagonal element to be 1 and off-diagonal element to be 0, it can easily be proved that solution exists only if all the variables are chosen from a field with characteristic 2. Note that the total number of variables used in this formulation is 15 and the size of F is only 5×5 . However, if the original line graph is used, the number of variables is 28 and the dimension of the adjacent matrix is 18×18 . Therefore, the total amount of reduction in problem dimension is 46% and in complexity is 92%.

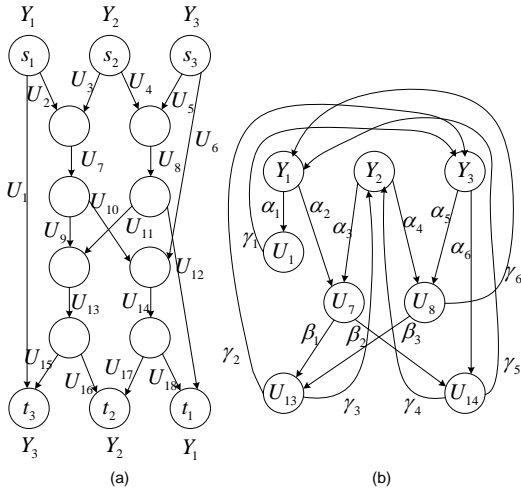


Fig. 6. A network that is solvable only over fields with characteristic 2

V. CONCLUSIONS

In this paper, reduced FDG is obtained from the original FDG by removing all the edges where network coding cannot

be performed and where network coding is not necessary. With reduced FDG, various capacity bounds of the network can be computed much more efficiently without any loss in the tightness. Two applications of reduced FDG are discussed in this paper, one is in the computation of LP bound and the other is in algebraic formulation of scalar linear network coding. In both cases, the reduced FDG significantly reduces the problem size and complexity. Note that the applications of the reduced FDG is not limited to the areas discussed in this paper. It may also help to identify the encoding complexity and simplify the code construction.

ACKNOWLEDGEMENT

The work of Xiaoli Xu and Yong Liang Guan was supported by the Advanced Communications Research Program DSOCL06271, a research grant from the Directorate of Research and Technology (DRTech), Ministry of Defence, Singapore. The work of Satyajit Thakor was partially supported by a grant from the University Grants Committee of the Hong Kong Special Administrative Region, China (Project No. AoE/E-02/08).

REFERENCES

- [1] R. Ahlswede and N. Cai and S.-Y. R. Li and R. W. Yeung, "Network Information Flow," *IEEE Trans. Inform Theory*, vol. 46, pp. 1204-1216, July 2000.
- [2] G. Kramer, *Directed Information for Channels with Feedback*. PhD thesis, Swiss Federal Institute of Technology, Zurich, 1998.
- [3] G. Kramer and S. Savari, "Edge-Cut Bounds on Network Coding Rates," *Journal of Netw. Syst. Manage.*, vol. 14, No. 1, pp. 49-67, 2006.
- [4] N. Harvey, R. Kleinberg, and A. Lehman, "On the capacity of information networks," *IEEE Trans. Inform Theory*, vol. 52, pp. 2345-2364, June 2006.
- [5] S. Thakor, A. Grant, and T. Chan, "Network coding capacity: a functional dependence bound," *Proc. 2009 IEEE Int. Symp. Inform. Theory*, (Seoul, Korea), pp. 263-267, June 28-July 3, 2009.
- [6] X. Yan, R. W. Yeung, and Z. Zhang, "The capacity region for multi-source multi-sink network coding," in *IEEE Int. Symp. Inform. Theory*, (Nice, France), pp. 116-120, Jun. 2007.
- [7] F. Matus, "Infinitely many information inequalities," in *IEEE Int. Symp. Inform. Theory*, 2007.
- [8] R. W. Yeung, *Information Theory and Network Coding*. New York: Springer, 2008.
- [9] R. Koetter and M. Médard, "An Algebraic Approach to Network Coding," *IEEE/ACM Transactions on Networking*, vol. 11, No. 5, October 2003.
- [10] S. U. Kamath, D. N. C. Tse and V. Anantharam, "Generalized Network Sharing Outer Bound and the Two-Unicast Problem", *Proc. of the IEEE International Symposium on Network Coding* (Beijing, China), 2011.
- [11] R. Dougherty, C. Freiling, and K. Zeger, "Insufficiency of linear coding in network information flow," *IEEE Trans. On Inf. Theory*, vol. 51, no. 8, August 2005.
- [12] W. Song, K. Cai, R. Feng and W. Rui, "Solving the two simple multicast network coding problem in time $O(|E|)$ ", *ICCSN 2011*.
- [13] S. Thakor, A. Grant and T. Chan, "On Complexity Reduction of the LP Bound Computation and Related Problems". *Proc. of the IEEE International Symposium on Network Coding* (Beijing, China), 2011.